Special Issue
"Progress of Theoretical and Computational Biophysics"

Vol. 16, pp. 295–303 (2019)
doi: 10.2142/biophysico.16.0_295

Biophysics and Physicobiology
https://www.jstage.jst.go.jp/browse/biophysico/

Regular Article

More efficient screening of protein-protein complex model structures for reducing the number of candidates

Kazuhiro Takemura and Akio Kitao

School of Life Science and Technology, Tokyo Institute of Technology, Meguro-ku, Tokyo 152-8550, Japan

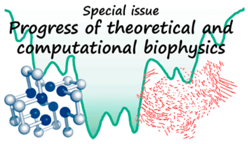
Received June 17, 2019; accepted August 1, 2019

Rigid-body protein-protein docking is very efficient in generating tens of thousands of docked complex models (decoys) in a very short time without considering structure change upon binding, but typical docking scoring functions are not necessarily sufficiently accurate to narrow these decoys down to a small number of plausible candidates. Flexible refinements and sophisticated evaluation of the decoys are thus required to achieve more accurate prediction. Since this process is time-consuming, an efficient screening method to reduce the number of decoys is necessary immediately following rigid-body dockings. We attempted to develop an efficient screening method by clustering decoys generated by the rigid-body docking ZDOCK. We introduced the three metrics ligand-root-mean-square deviation (L-RMSD), interface-ligand-RMSD (iL-RMSD), and the fraction of common contacts (FCC), and examined various ranges of cut-offs for clusters to determine the best set of clustering parameters. Although the employed clustering algorithm is simple, it successfully reduced the number of decoys. Using iL-RMSD with a cut-off radius of 8 Å, the number of decoys that contain at least one near-native model with 90% probability decreased from 4,808 to 320, a 93% reduction in the original number of decoys. Using FCC for the clustering step, the top 1,000 success rates, defined as the probability that the top 1,000 models contain at least one near-native structure, reached 97%. We conclude that the proposed method is very efficient in selecting a small number of decoys that include near-native decoys.

Key words: protein-protein docking, clustering, fraction of common contacts, complex structure prediction

Protein-protein interactions play central roles in biological process at the molecular level [1] and thus, structures of protein-protein complexes at atomic resolution provides valuable information for understanding the molecular mechanisms underlying these processes. Atomic-resolution structures of protein complexes are typically determined by X-ray crystallography, solution NMR, and cryo-electron microscopy, but studies are often time-consuming and sometimes very difficult. Therefore, computational approaches are very useful if they can predict protein-protein complex structures accurately and efficiently. To this end, the last 20 years or so have seen the development of protein-protein docking prediction methods and their increasingly wide use

Corresponding authors: Kazuhiro Takemura, School of Life Science and Technology, Tokyo Institute of Technology, 2-12-1 Ookayama, Meguro-ku, Tokyo 152-8550, Japan. e-mail: ktakemura@bio.titech.ac.jp; Akio Kitao, School of Life Science and Technology, Tokyo Institute of Technology, 2-12-1 Ookayama, Meguro-ku, Tokyo 152-8550, Japan. e-mail: akitao@bio.titech.ac.jp



◀ ***Significance*** ▶

We proposed an efficient screening method to decrease the number of protein-protein complex model structures (decoys) using a relatively simple clustering method. By applying our approach to the decoys generated by the rigid-body docking method, ZDOCK, we reduced the number of decoys by 93% compared to ZDOCK in terms of the number of decoys containing at least one near-native decoy with 90% probability. After clustering using the fraction of common contacts, the top 1,000 success rate (defined as the rate having at least one near-native model in the selected decoys) reached 97%.

[2–8]. Typical protein-protein docking techniques include rigid-body docking methods based on Fast Fourier transform (FFT) which efficiently generate complex model structures (so-called decoys) [9–14], flexible docking and structure refinement approaches aimed at understanding structural changes upon complex formation [15–19], and binding free energy calculations which enables the accurate evaluation of docking-generated decoys [20,21]. Each method offers different advantages in efficiency and accuracy and thus we propose using a procedure combining these different techniques (Fig. 1). The proposed procedure uses rigid-body docking to generate tens of thousands of decoys and to score-base rank the decoys, followed by clustering of decoys to narrow the number of candidates, and finally flexible structure refinement and binding free energy calculations to select the final predicted structures. Given the many rigid-body docking methods developed to date and their success in generating a set of decoys that includes structures similar to the native structures (called near-native decoys), it is essential to select near-native decoys from other generated structures by applying proper evaluation criteria. Over the past ten years we have developed a method, evERdock, to evaluate the decoys by calculating the binding free energies for decoys by combining a short all-atom molecular dynamics simulation with explicit solvent and solution theory in the energy representation [20–22]. Although evERdock was demonstrated to be applicable to hundreds of decoys, it remains time-consuming. Therefore, efficient screening of the decoys generated by rigid-body docking prior to applying evERdock is an important process in this procedure.

Protein-protein docking methods are often accompanied by post-processing steps to reduce the number of docking generated decoys by clustering based on different algorithms and metrics. ClusPro server [23,24] calculates all pair-wise interface ligand root-mean-square distances (iL-RMSDs) between the top 1,000 decoys generated by rigid-body docking where iL-RMSD is defined as the RMSD for the interface residues of the smaller protein after superposing the larger protein. In ClusPro, clustering is conducted based on the algorithm suggested by Daura, X., *et al.* [25]: a decoy is considered as a neighbor of another decoy if the iL-RMSD from the decoy is equal to or less than 10 Å. First, the decoy with the largest numbers of neighbors is considered as the center of the first cluster, then this cluster center and the neighbors are removed from the decoy pool. The remaining decoys with the largest number of neighbors is selected as the center of the next-ranked cluster, and the selected decoys are removed. This procedure is repeated for the remaining decoys until the specified number of cluster centers are selected. We call this procedure population-based clustering because clusters are rank-ordered based on the cluster population. HADDOCK [15] generates decoys using molecular dynamics, conducts population-based clustering, and ranks the clusters based on the HADDOCK docking score averaged over the top four decoys of each cluster. A recent update to HADDOCK introduced fraction of common contacts (FCC) as the clustering metric, calculated from residue-based inter-protein contact pairs which are common between two decoys [26]. A decoy is considered as a neighbor of another decoy if the FCC of the decoys is equal to or greater than 0.75. Clustering used in FRODOCK [11,27] involves assigning the top (highest-score) decoy as the center of the first cluster and the members of the first cluster are selected as the decoys if their ligand-RMSD (L-RMSD) from the top decoy is equal to or less than 5 Å. After removing the cluster members, the procedure is repeated until 10,000 clusters are obtained or all the decoys are clustered. This score-based clustering requires less computation compared to the population-based clustering employed in ClusPro and

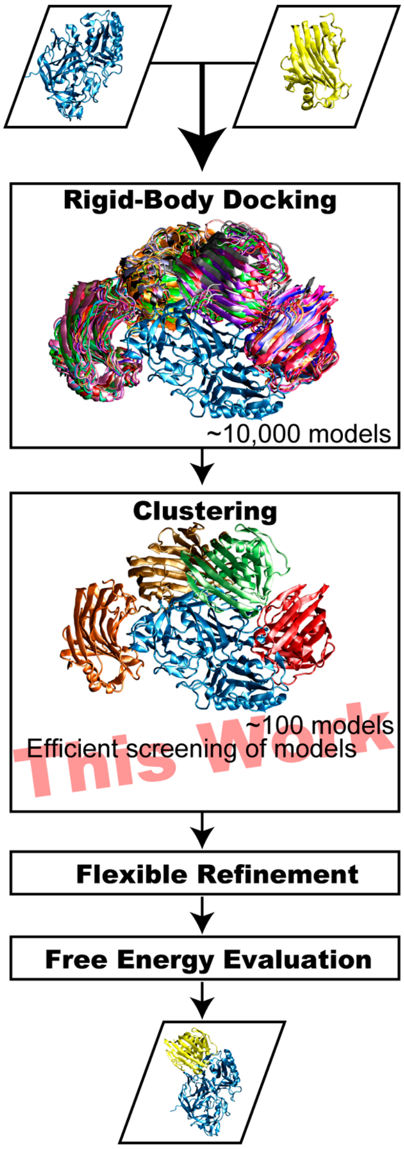


Figure 1 Flow chart for the structure prediction of protein-protein complexes considered in this study.

HADDOCK because not all the pairs of decoys are necessarily compared. The InterEvDock server [28] employs FRODOCK for the rigid-body docking and conducts a clustering with the FCC metric at a later stage. CyClus [29] performs hierarchical clustering and re-ranks the decoys generated by a rigid-body docking method using both docking and clustering scores.

In this work, we survey better clustering metrics and cut-off parameters so as to better screen decoys generated by rigid-body docking. For this purpose, we conduct decoy clustering, employing L-RMSD, iL-RMSD, and FCC as the clustering metrics. We show that score-based clustering with the iL-RMSD metric efficiently (93%) reduces the number of decoys with at least one near-native decoy with 90% probability and that top 1,000 success rates (where ‘success rate’ is defined as a rate with at least one near-native decoy) was 97% using score-based clustering with FCC.

Methods

Benchmark dataset of protein-protein complex structures

We examined the performance of various clustering metrics by using the protein-protein docking benchmark 5.0 [30] as a database of known protein-protein complex structures. Based on docking difficulty, this benchmark classifies target complexes into rigid, medium, and difficult classes comprising 151, 45, and 34 complexes, respectively. As mentioned in the Introduction, the purpose of this work is to efficiently reduce the number of decoys immediately following rigid-body docking, before refinement and evaluation at later stages. Protein flexibility is considered after the docking and clustering stages (Fig. 1) and thus here we focus mainly on complexes in the rigid class. The results shown below are obtained from the complexes in the rigid class unless otherwise specified. Also, we focus on hetero-oligomers in the benchmark set, resulting in 185 complexes which consist of 121 rigid, 37 medium, and 27 difficult complexes.

Decoy generation

Decoys were generated using the rigid-body protein-protein docking program ZDOCK 3.0.2 [10,12]. In ZDOCK 3, optimal translational positions for a given orientation of the smaller protein relative to the fixed larger protein are determined using an FFT-based method. A total of 54,000 decoys were generated by grid search of the rotational space at 6° increments.

For comparison, we employed another rigid-body protein-protein docking program FRODOCK 2.1 [11,27] with default settings. For a given translational position, FRODOCK performs a fast rotational search using spherical harmonics. After obtaining the optimized translational and rotational positions, FRODOCK conducts score-based clustering with the L-RMSD metric and a 5 Å-clustering cut-off radius (R_C) as mentioned in the Introduction.

Re-ranking of generated decoys

The decoys generated by ZDOCK were re-ranked based on the following three methods. The first method is ZRANK 2 [31,32], which is a docking refinement program developed to provide fast and accurate rescoring of models (hereafter denoted as ZDOCK/ZRANK). The second is the clustering/re-ranking method CyClus [29] which rapidly clusters and re-ranks decoys using a cylindrical approximation of the protein-protein complex interface and hierarchical clustering (denoted as ZDOCK/CyClus). In the third method, we re-ranked the decoys using the atomic pair potential proposed by Tobi (denoted as ZDOCK/Tobi) [33].

Clustering procedure

We conducted score-based clustering using the aforementioned scores and the three metrics L-RMSD, iL-RMSD, and FCC. L-RMSD is the simplest metric to calculate because receptor proteins are usually fixed during rigid-body dockings and thus L-RMSD can be calculated without further structure superposition. Calculation of iL-RMSD requires assignments of interface residues and superposition of the interface residues between a pair of decoys because iL-RMSD is the L-RMSD of the interface residues. Here, the interface residues are defined as those having at least one heavy atom within 10 Å of any heavy atom of the partner protein [10]. The FCC clustering metric is efficient for clustering protein-protein docking decoys [26]. To calculate FCC, a pair of residues from two distinct proteins are considered to be in contact if any of their atoms are within 5 Å [34]. We examined different cut-off values to assign decoys to clusters; the clustering with L-RMSD and iL-RMSD used R_C values from 5 to 15 Å with increments of 1 Å, and those with FCC used cut-off fractions (f_C) from 0.2 to 0.8 with 0.05 increments.

After clustering, the decoys selected as the cluster centers are regarded as the cluster representatives and they are the only decoys considered in the following analysis, which means that the number of decoys is highly reduced in this step. The selected decoys are rank-ordered by the scores of the cluster centers as described above. We also conducted re-ranking of the selected decoys based on the average ZDOCK score of all decoys in each cluster and the average ZDOCK score of the top 10 ranked decoys in each cluster.

The population-based clustering requires calculation of all the pair-wise RMSDs or FCCs, which takes much longer computational time to conduct clustering of many decoys. To examine various combinations of cut-off values and metrics for clustering many (54,000) decoys, we decided to use the score-based clustering in this study.

Evaluation of the clustering results

To evaluate the decoys, we followed the criteria used in Critical Assessment of PRedicted Interactions (CAPRI) [35]. In this study, a decoy with an acceptable or better quality according to the CAPRI criteria is called a near-native decoy and thus should satisfy one of the following two conditions: (i) the fraction of native contacts (f_{nat}) is at least 0.3 and (ii) f_{nat} is at least 0.1 and L-RMSD is no more than 10 Å (or the interface RMSD is no more than 4.0 Å). The cluster is called a near-native cluster if the cluster center is the near-native decoy.

After efficient screening of the decoys, our goal is to construct a set of a small number of decoys that contains at least one near-native decoy with high probability. We evaluate the screening efficiency by introducing the numbers of decoys containing at least one near-native decoy with probabilities of 80% and 90% as $N_{80\%}$ and $N_{90\%}$, respectively, among complexes in the benchmark set. In other words, if the top $N_{90\%}$ decoys are selected for a certain complex, we can expect that at least one near-native decoy is included in the selected decoys in 90% of the cases. We also evaluated the results using a second property by examining the success rate defined as the percentage having at least one near-native model in the top N ranked models among the examined complexes.

F. Computational time

The computations were conducted using a single core Intel Xeon (R) CPU E3-1240, E3-1620, E5-1660, or E5-2695 for a given complex. The computation times in this study for 185 complexes are as follows: rigid-body dockings by ZDOCK and FRODOCK took 2.2±1.2 hours and 1.0±1.0 hours, respectively, re-rankings of ZDOCK decoys by ZRANK, CyClus, and Tobi required 1.2±1.1 hours, 4.5±1.0 minutes, and 26±4.5 minutes, respectively, and computational times for score-based clustering of the ZDOCK-generated decoys with L-RMSD (with a cut-off radius 9 Å), iL-RMSD (cut-off radius 8 Å), FCC (cut-off fraction 0.3) were 15±13 minutes, 29±27 minutes, and 11±4.4 minutes, respectively. The score-based clustering methods proposed in this study required more computation time than CyClus but are acceptable. CyClus was developed in our group and optimized for our computer environment. Note that we used executable ZDOCK, FRODOCK, and ZRANK files distributed by the original authors, whereas we wrote VMD [36] scripts for Tobi and score-based clustering methods. The codes for the score-based clustering methods can be further optimized if necessary, however, we did not write optimized program in this work because the calculations with VMD scripts could be completed within a reasonable computational time frame.

Results and Discussion

Parameter dependence of the clustering results

As described in the Methods section, we employed the three metrics L-RMSD, iL-RMSD, and FCC to conduct score-based clustering 54,000 decoys generated by ZDOCK 3.0.2 [10,12]. As R_C increased or f_C decreased, the number of decoys in each cluster increased and the number of cluster decreased (Fig. 2A). iL-RMSD clustering returned more decoys in each cluster than L-RMSD clustering because iL-RMSD clustering focuses only on the interface residues and iL-RMSD tends to be smaller than L-RMSD for a given decoy. As the number of clusters decreased, the number of near-native decoys decreased (Fig. 2B). Also, the fraction of near-native decoys (the number of near-native decoys divided by the number of clusters) decreased after clustering (Fig. 2C). Overall Figure 2 shows the reasonable relationship between cut-off value and clustering results. In the parameter range examined, FCC clustering generated more near-native decoys than the other approaches. Note that FCC does not show a linear relationship with RMSD. As an exam-

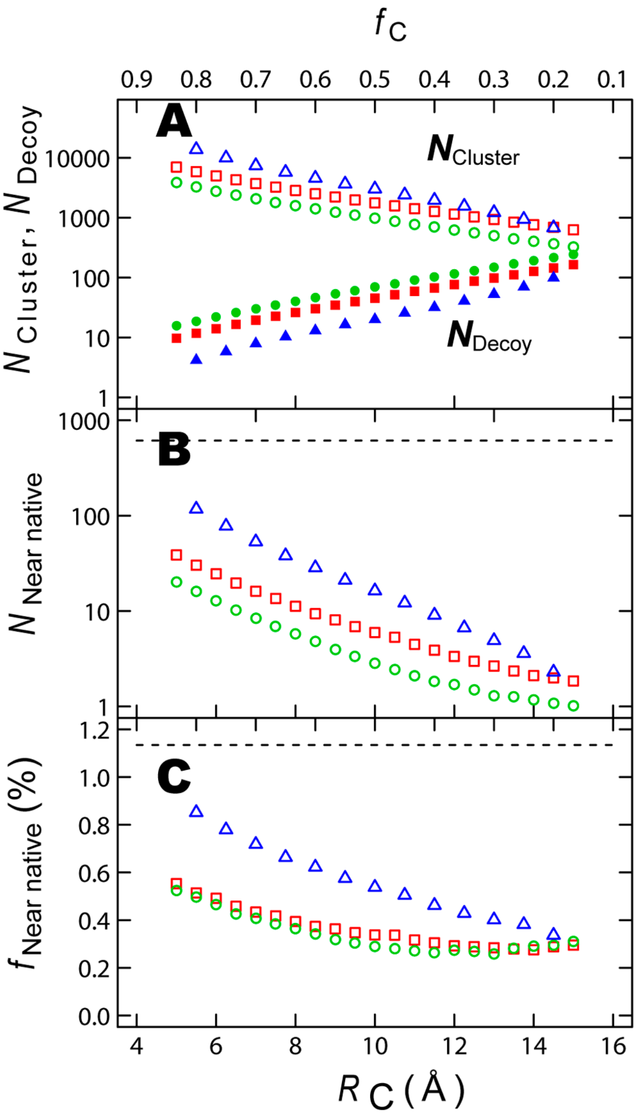


Figure 2 Cut-off radius R_C (fraction f_C) dependence of (A) the numbers of clusters (open symbols), decoys in each cluster (filled symbols), (B) near-native decoys, and (C) fractions of near-native decoys among the selected decoys. Red square, green circle, and blue triangle represent the results of clustering with L-RMSD, iL-RMSD, and FCC, respectively. Broken lines in (B) and (C) indicate the number of decoys and the fraction of near-native decoys before clustering, respectively.

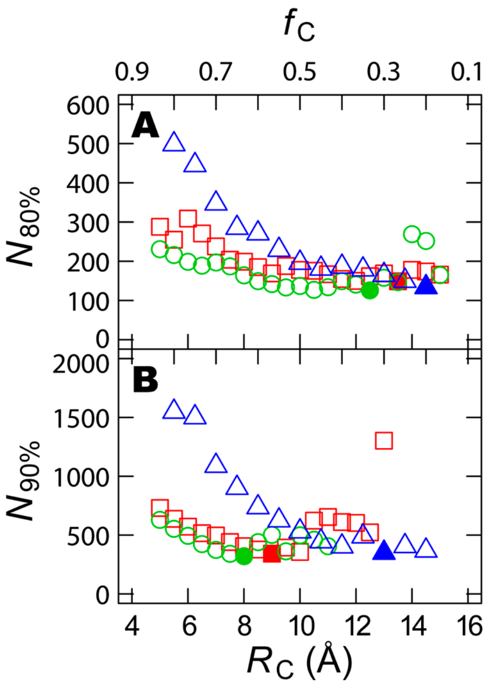


Figure 3 Cut-off radius (fraction) dependence of the number of clusters containing at least one near-native cluster at (A) 80% and (B) 90% probability. Red squares, green circles, and blue triangles represent the results obtained using L-RMSD, iL-RMSD, and FCC clustering, respectively. Filled symbols indicate the cut-off values which provide the smallest numbers of clusters.

ple of this non-linear relationship, the values of the fraction of native contacts used to define acceptable, medium, and high quality in CAPRI are 0.1, 0.3, and 0.5, respectively, whereas those of L-RMSD are 10, 5, and 1 Å, respectively. The fractions after clustering are lower than the fraction of near-native decoys (the number of near-native decoys among the selected decoys divided by 54,000). As mentioned in the Introduction, the purpose of this work is the efficient reduction of the number of decoys that contain at least one near-native decoy rather than the enrichment of near-native decoys among other decoys. Therefore, the current approach is not suitable for enrichment purposes.

We calculated $N_{80\%}$ (Fig. 3A) and $N_{90\%}$ (Fig. 3B) to examine the efficiency of the clustering in reducing the number of decoys that contain at least one near-native decoy. As R_C increases, $N_{80\%}$ and $N_{90\%}$ first decreases and then increases, which means that $N_{80\%}$ and $N_{90\%}$ have a minimum value at optimum R_C. FCC clustering also shows a similar trend. When top-ranked decoys generated by ZDOCK are not near-native and resemble each other, clustering of these decoys into single cluster improves rankings of the near-native decoys. Such effects are enhanced by increasing R_C or decreasing F_C, which results in decrease of $N_{80\%}$ and $N_{90\%}$. However, the use of larger R_C or smaller F_C leaded more near-native decoys to be the members of the non-near-native cluster; eventually, $N_{80\%}$ and $N_{90\%}$ increased. These effects determine the optimal values of R_C and F_C. We selected 9.0 Å, 8.0 Å, and 0.3 as the optimal cut-off values to achieve the lowest $N_{90\%}$ for clustering with L-RMSD, iL-RMSD, and FCC, respectively. In addition, we also selected 13.5 Å, 12.5 Å, and 0.2 to achieve lowest $N_{80\%}$ with L-RMSD, iL-RMSD, and FCC, respectively. Hereafter, we call these clustering methods Lr9, iLr8, Fc3, Lr13.5, iLr12.5, and Fc2 which represent the combination of the metric and the cut-off value.

Comparison of docking/clustering performance

Table 1 summarizes the docking/clustering performances obtained using Lr9, iLr8, Fc3, Lr13.5, iLr12.5, and Fc2. For comparison, we also show the docking and docking/re-ranking results obtained using several methods (see the Methods section for details). We mainly focus on the results for the rigid class but also show the results obtained for all complexes of all classes in the benchmark in parentheses. Compared to the ZDOCK results, the value of $N_{90\%}$ obtained

Table 1 Summary of docking/clustering performance

Method	$N_{80\%}$	$N_{90\%}$	Top N success rate (%)			
			N=10	N=100	N~1000[a]	All[b]
ZDOCK	1552 (3306)	4808 (18207)	28 (24)	50 (46)	77 (69)	98 (94)
ZDOCK/ZRANK	364 (1641)	2036 (11567)	37 (29)	67 (53)	87 (78)	98 (94)
ZDOCK/CyClus	362 (732)	1083 (2679)	36 (31)	67 (59)	89 (82)	98 (94)
ZDOCK/Tobi	557 (1645)	2304 (10204)	31 (26)	60 (52)	85 (77)	98 (94)
FRODOCK	537 (1350)	1973 (–)	36 (30)	64 (56)	84 (78)	94 (89)
ZDOCK/Lr9	169 (342)	342 (1210)	44 (36)	75 (65)	93 (86)	98 (92)
ZDOCK/iLr8	164 (320)	320 (1137)	43 (36)	74 (65)	95 (89)	98 (92)
ZDOCK/Fc3	165 (335)	350 (1300)	43 (36)	73 (64)	97 (89)	98 (91)
ZDOCK/Lr13.5	149 (–)	– (–)	41 (35)	74 (63)	88 (79)	88 (80)
ZDOCK/iLr12.5	126 (–)	– (–)	45 (37)	74 (62)	88 (77)	88 (77)
ZDOCK/Fc2	134 (479)	368 (–)	46 (37)	74 (63)	93 (83)	93 (83)

[a]After clustering, the numbers of clusters for some complexes are less than 1,000. [b]All the 54,000 decoys are considered in ZDOCK. In FRODOCK, the numbers of decoys are at most 10,000. The average numbers of the cluster after clustering are 2222, 1588, 1224, 845, 559, and 681 for Lr9, iLr8, Fc3, Lr13.5, iLr12.5, and Fc2, respectively. Values in parentheses are the results for complexes in all classes in the benchmark.

by iLr8 decreased from 4,808 to 320 (93% reduction). The decreases achieved with the other metrics (Lr9 and Fc3) were also significantly better than that by ZDOCK but were slightly less than that achieved with iLr8. Other docking and re-ranking methods also decreased the required number of decoys; however, iLr8 outperformed all the other methods to obtain the smallest $N_{90\%}$. The clustering with iLr12.5 obtained the smallest $N_{80\%}$.

As a typical evaluation of docking methodologies, we also show the top N (N=10, 100, and 1,000) success rates in Table 1, which show the expected percentage (probability) that at least one near-native decoy is included in the top N decoys. The top 10 and 100 success rates were highest in the Lr9 result (44% and 75%, respectively) whereas the top 1,000 success rate was highest in the Fc3 results (97%). The numbers of clusters generated by Lr9, iLr8, Fc3, Lr13.5, iLr12.5, and Fc2 are on average 2,222, 1,588, 1,224, 845, 559, and 681, respectively, and less than 1,000 for some complexes. Thus, the top 1,000 success rates contain results from less than 1,000 clusters. Although the fraction of near-native decoys decreased (Fig. 2C), the clusters obtained by Lr9, iLr8, and Fc3 still contain near-native decoys in most of the benchmark complexes (rightmost column in Table 1). On the other hand, the clusters obtained by Lr13.5, iLr12.5, and Fc2 do not contain near-native decoys in some cases, which results in lower top 1,000 success rates than those obtained by Lr9, iLr8, Fc3. Therefore, we focus on Lr9, iLr8, and Fc3 in the following analysis.

Overall, our approaches successfully reduced $N_{90\%}$ from 4,808 in ZDOCK to a few hundred. Also, we achieved a very high top 1,000 success rate (97%) using Fc3. To our knowledge, no docking methods with this high success rate have been reported to date.

Clustering of the decoys ranked by other methods

Our score-based clustering approaches, Lr9, iLr8, and Fc3 efficiently screen ZDOCK-generated decoys. We also applied the Lr9, iLr8, and Fc3 methods to the decoys re-ranked by ZDOCK/ZRANK, ZDOCK/CyClus, ZDOCK/Tobi and those generated by FRODOCK (Table 2). We obtained similar or even better performance when we applied the Lr9, iLr8, and Fc3 methods to the decoys re-ranked by ZRANK and CyClus. For instance, $N_{80\%}$=113 with ZRANK/iLr8 is smallest of any result obtained so far, and the $N_{90\%}$ and the top N success rates are similar to those shown in Table 1. ZDOCK/Tobi improved the ZDOCK results, but further clustering provided results that were slightly worse. The improvement obtained with ZDOCK/Tobi did not outperform ZRANK or CyClus in terms of $N_{90\%}$ or the top N success rates. Since the cut-off values (9.0 Å, 8.0 Å, and 0.3) were optimized for the decoys generated by ZDOCK, additional parameter tunings might improve performance for decoys generated by other ranking methods. Lr9, iLr8, and Fc3 did not improve the FRODOCK results probably because FRODOCK provided the results of clustering with L-RMSD and R_C=5 Å and thus further clustering did not improve the results.

Effect of different cluster ranking methods

The clustering conducted thus far ranked the clusters according to the score of the decoy selected as the cluster center. Some docking methods rank the clusters differently. For example, ClusPro [23] employs population-based clustering that ranks the clusters by the number of decoys in each cluster. In contrast, HADDOCK [15] ranks the clusters based on the average score of the top four decoys of each cluster. We attempted variations of the clustering based

Table 2 Docking/clustering performance combined with other ranking methods

Method	$N_{80\%}$	$N_{90\%}$	Top N success rate (%)			
			N=10	N=100	N=1000	All[a]
ZDOCK/ZRANK	364 (1641)	2036 (11567)	37 (29)	67 (53)	87 (78)	98 (94)
ZRANK/Lr9	120 (472)	525 (2842)	46 (36)	77 (64)	92 (85)	97 (91)
ZRANK/iLr8	113 (455)	440 (–)	46 (36)	77 (63)	93 (84)	97 (89)
ZRANK/Fc3	109 (409)	379 (–)	46 (36)	78 (64)	93 (84)	96 (86)
ZDOCK/CyClus	362 (732)	1083 (2679)	36 (31)	67 (59)	89 (82)	98 (94)
CyClus/Lr9	176 (403)	412 (2764)	42 (35)	72 (62)	94 (86)	98 (90)
CyClus/iLr8	158 (338)	412 (1488)	41 (34)	72 (62)	96 (89)	98 (91)
CyClus/Fc3	154 (394)	396 (–)	40 (33)	73 (63)	94 (87)	96 (89)
ZDOCK/Tobi	557 (1645)	2304 (10204)	31 (26)	60 (52)	85 (77)	98 (94)
Tobi/Lr9	293 (601)	831 (4516)	34 (28)	65 (57)	93 (84)	98 (90)
Tobi/iLr8	254 (509)	702 (–)	34 (28)	64 (56)	93 (84)	97 (89)
Tobi/Fc3	232 (524)	763 (–)	32 (26)	63 (55)	93 (83)	93 (85)
FRODOCK	537 (1350)	1973 (–)	36 (30)	64 (56)	84 (78)	94 (89)
FRODOCK/Lr9	688 (–)	– (–)	36 (29)	61 (54)	84 (75)	86 (77)
FRODOCK/iLr8	670 (–)	– (–)	35 (28)	60 (54)	84 (73)	86 (75)
FRODOCK/Fc3	318 (949)	1326 (–)	37 (30)	66 (58)	88 (80)	91 (82)

Table 3 Docking/clustering performance with different cluster ranking methods

Clustering	Re-ranking	$N_{80\%}$	$N_{90\%}$	Top N success rate (%)		
				N=10	N=100	N=1000
Lr9	–	169 (342)	342 (1210)	44 (36)	75 (65)	93 (86)
	Top 10	286 (639)	669 (1781)	45 (36)	70 (60)	96 (86)
	Average	477 (877)	786 (1978)	8 (6)	36 (27)	95 (83)
	N_{Decoys}	153 (426)	619 (1893)	39 (31)	70 (60)	93 (86)
iLr8	–	164 (320)	320 (1137)	43 (36)	74 (65)	95 (89)
	Top 10	310 (541)	541 (1319)	45 (35)	71 (62)	94 (87)
	Average	465 (742)	730 (1624)	5 (4)	26 (21)	95 (86)
	N_{Decoys}	149 (400)	536 (1270)	31 (24)	72 (60)	94 (88)
Fc3	–	165 (335)	350 (1300)	43 (36)	73 (64)	97 (89)
	Top 10	223 (543)	471 (1243)	46 (37)	72 (62)	94 (87)
	Average	446 (651)	651 (1468)	10 (8)	34 (26)	93 (85)
	N_{Decoys}	160 (409)	409 (1322)	26 (19)	73 (62)	97 (89)

on the average score of the top 10 decoys of each cluster (Top 10 in Table 3), the average score of all the decoys in each cluster (Average), and the number of members in each cluster (N_{Decoys}).

Top 10 and Average did not improve the results in all cases as shown by the increases of $N_{80\%}$ and $N_{90\%}$ and the decrease in most of the top N success rates compared to the Lr9, iLr8, Fc3 results shown in Table 1. The average numbers of decoys in each near-native cluster after Lr9, iLr8, and Fc3 is 101, 142, and 154, respectively, and are higher than those in the non-near-native cluster (24, 34, and 43, respectively). Since the near-native clusters contain many non-near-native decoys with low scores, the average score is lowered by averaging. Such effects were prominently visible in the results using the average score of all the decoys. Interestingly, N_{Decoys} provided smaller $N_{80\%}$ and $N_{90\%}$ values than those obtained using the average scores. The partial success of N_{Decoys} is related to the fact that the near-native clusters tend to have a larger number of decoys. In summary, these trials did not considerably improve the results.

We further focus on the aforementioned tendency that the near-native clusters have a larger number of decoys. For this purpose, we investigated N_{Decoys} averaged over all, near-native, and top 10 clusters for each of complexes ($\langle N_{\text{Decoys}} \rangle$). A histogram of $\langle N_{\text{Decoys}} \rangle$ over 121 complexes was calculated and shown in Figure 4. The near-native clusters tend to have smaller $\langle N_{\text{Decoys}} \rangle$ than the top 10 clusters. The near-native clusters have more decoys than other clusters on average as shown before, but they have a smaller number of decoys in some cases. In these cases, the re-ranking by the N_{Decoys} is not suitable. The top 10 clusters contain many decoys, indicating that many of them are similar to the top-ranked decoys. When the top-ranked decoys are not near-native, the score-based clustering methods conducted in this study effectively decrease the number of non-near-native decoys and improve the ranking of the near-native decoys. The use of optimal clustering cut-offs maximizes such effects and contributes to the successful results obtained in this study.

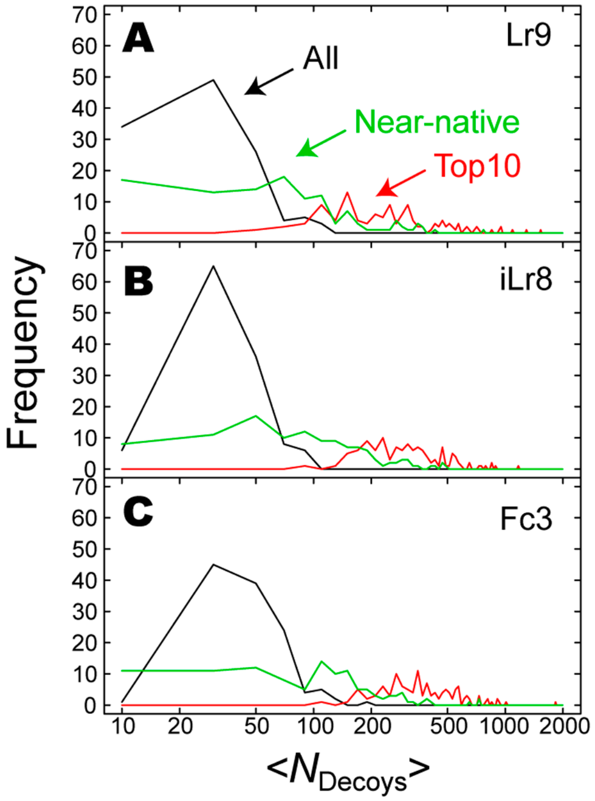


Figure 4 Histogram of the average number of decoys ($\langle N_{\text{Decoys}} \rangle$) with a bin size of 20. The numbers of decoys were averaged over all (black), the top 10 (red), and the near-native (green) clusters using (A) Lr9, (B) iLr8, and (C) Fc3.

Conclusion

Protein-protein complex structure prediction remains a challenging problem. Considering the docking procedure shown in Figure 1, an efficient method is required for reducing the number of candidate models after decoy generation to reduce the expected computation time of flexible refinement and free energy evaluation. This work proposed a simple but very efficient clustering approach to achieve this purpose. Using iLr8, $N_{90\%}$ decreased from 4,808 to 320, which is a 93% reduction in the number of decoys, and using Fc3, the top 1,000 success rate was as high as 97%.

Although we obtained promising results, further parameter tunings may improve this method. Possible modifications include the choice of rigid-body docking software, a combination of RMSD and FCC to distinguish decoys for clustering, and re-ranking using consensus selections after clustering. Since our score-based clustering approach with iL-RMSD, L-RMSD, or FCC successfully reduces the number of decoys, the following flexible refinements and free energy calculations will be able to treat all models after the clustering. For example, evERdock can evaluate all the selected decoys after the score-based clustering method because it has previously treated 300 decoys for multiple complexes [21]. We believe that this type of approach would improve the current status of protein-protein complex structure predictions.

Acknowledgment

This research was supported by MEXT/JSPS KAKENHI (No. JP17KT0026 and JP19H03191) to A. K., MEXT Priority Issues on Post-K Computer Projects "Building Innovative Drug Discovery Infrastructure through Functional Control of Biomolecular System" to A. K. The computations were partly performed using the supercomputers at the RCCS, The National Institute of Natural Science, and ISSP, The University of Tokyo. This research also used computational resources of the K computer provided by the RIKEN Advanced Institute for Computational Science through the HPCI System Research project (Project ID: hp150270, hp160207, hp170254, hp180201, and hp190181).

Conflicts of Interest

K. T. and A. K. declare that they have no conflict of interest.

Author Contributions

K. T. and A. K. directed the entire project and co-wrote the manuscript. K. T. carried out calculations.

References

[1] Jones, S. & Thornton, J. M. Principles of protein-protein interactions. *Proc. Natl. Acad. Sci. USA* **93**, 13–20 (1996).

[2] Vajda, S. & Kozakov, D. Convergence and combination of methods in protein-protein docking. *Curr. Opin. Struct. Biol.* **19**, 164–170 (2009).

[3] Lensink, M. F. & Wodak, S. J. Docking, scoring, and affinity prediction in CAPRI. *Proteins* **81**, 2082–2095 (2013).

[4] Vakser, I. A. Protein-protein docking: from interaction to interactome. *Biophys. J.* **107**, 1785–1793 (2014).

[5] Xue, L. C., Dobbs, D., Bonvin, A. M. & Honavar, V. Computational prediction of protein interfaces: A review of data driven methods. *FEBS Lett.* **589**, 3516–3526 (2015).

[6] Lensink, M. F., Velankar, S. & Wodak, S. J. Modeling protein-protein and protein-peptide complexes: CAPRI 6th edition. *Proteins* **85**, 359–377 (2017).

[7] Halperin, I., Ma, B., Wolfson, H. & Nussinov, R. Principles of docking: An overview of search algorithms and a guide to scoring functions. *Proteins* **47**, 409–443 (2002).

[8] Smith, G. R. & Sternberg, M. J. Prediction of protein-protein interactions by docking methods. *Curr. Opin. Struct. Biol.* **12**, 28–35 (2002).

[9] Gabb, H. A., Jackson, R. M. & Sternberg, M. J. E. Modelling protein docking using shape complementarity, electrostatics and biochemical information. *J. Mol. Biol.* **272**, 106–120 (1997).

[10] Chen, R. & Weng, Z. P. Docking unbound proteins using shape complementarity, desolvation, and electrostatics. *Proteins* **47**, 281–294 (2002).

[11] Garzon, J. I., Lopez-Blanco, J. R., Pons, C., Kovacs, J., Abagyan, R., Fernandez-Recio, J., *et al.* FRODOCK: a new approach for fast rotational protein-protein docking. *Bioinformatics* **25**, 2544–2551 (2009).

[12] Pierce, B. G., Hourai, Y. & Weng, Z. P. Accelerating Protein Docking in ZDOCK Using an Advanced 3D Convolution Library. *PLoS One* **6**, e24657 (2011).

[13] Ritchie, D. W. & Venkatraman, V. Ultra-fast FFT protein docking on graphics processors. *Bioinformatics* **26**, 2398–2405 (2010).

[14] Padhorny, D., Kazennov, A., Zerbe, B. S., Porter, K. A., Xia, B., Mottarella, S. E., *et al.* Protein-protein docking by fast generalized Fourier transforms on 5D rotational manifolds. *Proc. Natl. Acad. Sci. USA* **113**, E4286–4293 (2016).

[15] Dominguez, C., Boelens, R. & Bonvin, A. M. HADDOCK: a protein-protein docking approach based on biochemical or biophysical information. *J. Am. Chem. Soc.* **125**, 1731–1737 (2003).

[16] Gray, J. J., Moughon, S., Wang, C., Schueler-Furman, O., Kuhlman, B., Rohl, C. A., *et al.* Protein-protein docking with simultaneous optimization of rigid-body displacement and side-chain conformations. *J. Mol. Biol.* **331**, 281–299 (2003).

[17] Zacharias, M. Protein-protein docking with a reduced protein model accounting for side-chain flexibility. *Protein Sci.* **12**, 1271–1282 (2003).

[18] Chaudhury, S. & Gray, J. J. Conformer selection and induced fit in flexible backbone protein-protein docking using computational and NMR ensembles. *J. Mol. Biol.* **381**, 1068–1087 (2008).

[19] Li, X., Moal, I. H. & Bates, P. A. Detection and refinement of encounter complexes for protein-protein docking: taking account of macromolecular crowding. *Proteins* **78**, 3189–3196 (2010).

[20] Shinobu, A., Takemura, K., Matubayasi, N. & Kitao, A. Refining evERdock: Improved selection of good protein-protein complex models achieved by MD optimization and use of multiple conformations. *J. Chem. Phys.* **149**, 195101 (2018).

[21] Takemura, K., Matubayasi, N. & Kitao, A. Binding free energy analysis of protein-protein docking model structures

by evERdock. *J. Chem. Phys.* **148**, 105101 (2018).

[22] Takemura, K., Guo, H., Sakuraba, S., Matubayasi, N. & Kitao, A. Evaluation of protein-protein docking model structures using all-atom molecular dynamics simulations combined with the solution theory in the energy representation. *J. Chem. Phys.* **137**, 215105 (2012).

[23] Kozakov, D., Hall, D. R., Xia, B., Porter, K. A., Padhorny, D., Yueh, C., *et al.* The ClusPro web server for protein-protein docking. *Nat. Protoc.* **12**, 255–278 (2017).

[24] Kozakov, D., Beglov, D., Bohnuud, T., Mottarella, S. E., Xia, B., Hall, D. R., *et al.* How good is automated protein docking? *Proteins* **81**, 2159–2166 (2013).

[25] Daura, X., Gademann, K., Jaun, B., Seebach, D., van Gunsteren, W. F. & Mark, A. E. Peptide folding: When simulation meets experiment. *Angew. Chem. Int. Ed.* **38**, 236–240 (1999).

[26] Rodrigues, J. P., Trellet, M., Schmitz, C., Kastritis, P., Karaca, E., Melquiond, A. S., *et al.* Clustering biomolecular complexes by residue contacts similarity. *Proteins* **80**, 1810–1817 (2012).

[27] Ramirez-Aportela, E., Lopez-Blanco, J. R. & Chacon, P. FRODOCK 2.0: fast protein-protein docking server. *Bioinformatics* **32**, 2386–2388 (2016).

[28] Yu, J., Vavrusa, M., Andreani, J., Rey, J., Tuffery, P. & Guerois, R. InterEvDock: a docking server to predict the structure of protein-protein interactions using evolutionary information. *Nucleic Acids Res.* **44**, W542–549 (2016).

[29] Omori, S. & Kitao, A. CyClus: a fast, comprehensive cylindrical interface approximation clustering/reranking method for rigid-body protein-protein docking decoys. *Proteins* **81**, 1005–1016 (2013).

[30] Vreven, T., Moal, I. H., Vangone, A., Pierce, B. G., Kastritis, P. L., Torchala, M., *et al.* Updates to the Integrated Protein-Protein Interaction Benchmarks: Docking Benchmark Version 5 and Affinity Benchmark Version 2. *J. Mol. Biol.* **427**, 3031–3041 (2015).

[31] Pierce, B. & Weng, Z. P. ZRANK: Reranking protein docking predictions with an optimized energy function. *Proteins* **67**, 1078–1086 (2007).

[32] Pierce, B. & Weng, Z. P. A combination of rescoring and refinement significantly improves protein docking performance. *Proteins* **72**, 270–279 (2008).

[33] Tobi, D. & Elber, R. Distance-dependent, pair potential for protein folding: results from linear optimization. *Proteins* **41**, 40–46 (2000).

[34] Mendez, R., Leplae, R., De Maria, L. & Wodak, S. J. Assessment of blind predictions of protein-protein interactions: current status of docking methods. *Proteins* **52**, 51–67 (2003).

[35] Lensink, M. F. & Wodak, S. J. Docking and scoring protein interactions: CAPRI 2009. *Proteins* **78**, 3073–3084 (2010).

[36] Humphrey, W., Dalke, A. & Schulten, K. VMD: Visual molecular dynamics. *J. Mol. Graph.* **14**, 33–38 (1996).